

## Cromemco Software Update Service Note 68000 Cromix-1

Date: August 30, 1982

Product: CRO-DL and CRO-DS

Version.Release: 20.05

Date production of this version began: Aug. 27, 1982 on 8"  
Aug. 27, 1982 on 5"

First serial number with this version: 1-10012 on 8"  
1-10070 on 5"

### SUMMARY

This version of the Cromix™ Operating System is the first version configured for use with Cromemco's new Dual Processor Unit (DPU) and its associated 16 bit memory boards. The boards are the Memory Control Unit (MCU), and 265K and 512K Memory Storage Units (256MSU and 512MSU). These boards are described in the June 1982 Cromemco **68000 Board Family Instruction Manual**, part number 023-2016.

The 68000 Cromix Operating System gives you the ability to run programs for both the Z80 and 68000 microprocessors. You may thus take advantage of the power offered by the 68000 while still maintaining compatibility with the large software base established for the Z80.

This note describes the differences between the 68000 Cromix Operating System, version 20.05, and the Z80 Cromix Operating System, version 11.11. Most of the differences exist at the level of the operating system's interaction with the hardware and are not apparent at the user level. The June 1982 **Cromix Instruction Manual**, part number 023-4022, should be used as the primary reference for user operation.

To further supplement the **Cromix Instruction Manual**, copies of the following SUDS notes for the Cromix Operating System have been included in this package: **Cromix-4**, part number 023-9532, and **Cromix-5**, part number 023-9540.

Future versions of the Cromix Operating System will grow to take greater advantage of the 68000 microprocessor. Users are urged to subscribe to the Software Update Service (SUDS) to keep appraised of changes to the 68000 Cromix Operating System and to receive new releases of the software. To become a SUDS subscriber, contact your Cromemco dealer.

## **HARDWARE CONFIGURATION**

This version of the Cromix software requires at least the following boards for proper operation: a DPU, an MCU, one or more MSUs, and a disk controller board.

Any hard disk drives used with the system must be controlled by a WDI-II Winchester Disk Interface board, revision B or D. Failure to use the proper controller board will jeopardize the data on your hard disk.

## **NEW SOFTWARE**

### **Crogen68.bin version RB 00.00**

This utility generates a 68000 Cromix operating system. It allows you to add or delete system drivers and to select various options when you configure the system. Crogen68 is located in the /gen directory.

This 68000 version of Crogen operates in the same manner as the Z80 version, which is documented in the **Cromix Instruction Manual**.

### **Cromix.sys version 20.05**

This file contains the 68000 operating system configured by Crogen68. During the boot procedure, this file is read into the system portion of memory.

### **Ecc.bin version RB 00.02**

Ecc is a 68000 Cromix utility used with the Cromemco 256MSU and 512MSU memory boards, which have the ability to perform memory error detection and correction. This program allows a privileged user to enable or disable memory error correction hardware and to display on the console the type and location of memory errors. Error correction is disabled when the system is reset and must be normally reenabled.

Ecc has the following command formats and capabilities:

- Ecc on**      Turns on error correction.
- Ecc off**     Turns off error correction.
- Ecc**         Reports if error correction is off or on.
- Ecc -e**      Reports any errors since the last Ecc -e. If there are no errors, nothing is displayed on the console.

If memory errors are suspected, you may want to accumulate an error history. The command file `/cmd/logerr.cmd` is provided for this purpose. It periodically checks whether any memory errors have occurred since the last check. Errors are written to the file `/etc/msu_errs`.

Generate an error history by typing on your console the command

**logerr nn &**

where **nn** is the desired number of seconds between each Ecc check of the error status. The ampersand places the command file execution in background.

In this version of the Cromix Operating System, error correction **MUST** be turned off when using Hdtest or any other program which directly accesses the hard disk (i.e., programs which don't use Cromix system calls). Disk related Cromix system calls disable error correcting for their duration. Failure to turn error correction off while performing DMA to a hard disk will jeopardize the data on the hard disk.

**Init version 02.76**

The 68000 Cromix software includes a new version of Init, which has two enhancements:

The disk RPM test now works with MSUs when initializing hard or floppy disks. MSU error correction is automatically disabled while initializing a hard disk, preventing conflicts between the error correction and DMA.

CNTRL-C characters are now trapped by Init, making it impossible to interrupt Init once started. Previously, when initializing a hard disk, the alternate track table would be destroyed if the program received a CNTRL-C character during initialization.

### 68000 Abort Codes

When the 68000 encounters an error condition, it generates an error code and begins exception processing based on the type of error. A message such as

Vector interrupt - 03 4E720002 00394E73 00000000 30740002

will be displayed on the console. If this happens type **CNTRL-C**. In many cases, this will return control to the operating system. If, however, the processor was in supervisor mode, **CNTRL-C** will have no affect and the system will have to be reset.

The two digits at the far left of the numbers above, 03 are the error code. Look up this code in the following table and take the appropriate debugging action.

The hexadecimal numbers to the right of the error code represent the contents of the top 16 bytes of the system stack at the time the error was encountered. This information is pushed on the system stack during exception processing and may be useful in debugging. The first two bytes represent the contents of the status register. The next long word represents the contents of the program counter.

The following table defines run time errors generated by the 68000:

Vector number	Dec	Address Hex	Assignment
0	0	000	Reset: initial system stack pointer
1	4	004	Reset: initial program counter
2	8	008	Bus Error
3	12	00C	Address Error
4	16	010	Illegal Instruction
5	20	014	Zero Divide
6	24	018	CHK Instruction
7	28	01C	TRAPV Instruction
8	32	020	Privilege Violation
9	36	024	Trace
10	40	028	Line 1010 Emulator
11	44	02c	Line 1111 Emulator

### **68000 Memory Usage**

The addressing capability of the 68000 and the availability of new higher-capacity memory boards has lead to a vertical arrangement of 68000 Cromix memory. That is, memory is treated as one block in which memory addresses increase from the bottom to the top of memory. Z80 Cromix memory, on the other hand, is arranged horizontally in banks of 64K bytes.

The 68000 Cromix Operating System has been adapted from the Z80 Cromix Operating System, so some sections are written in Z80 code and are executed by the Z80. Remaining sections are written in 68000 code and are executed by the 68000.

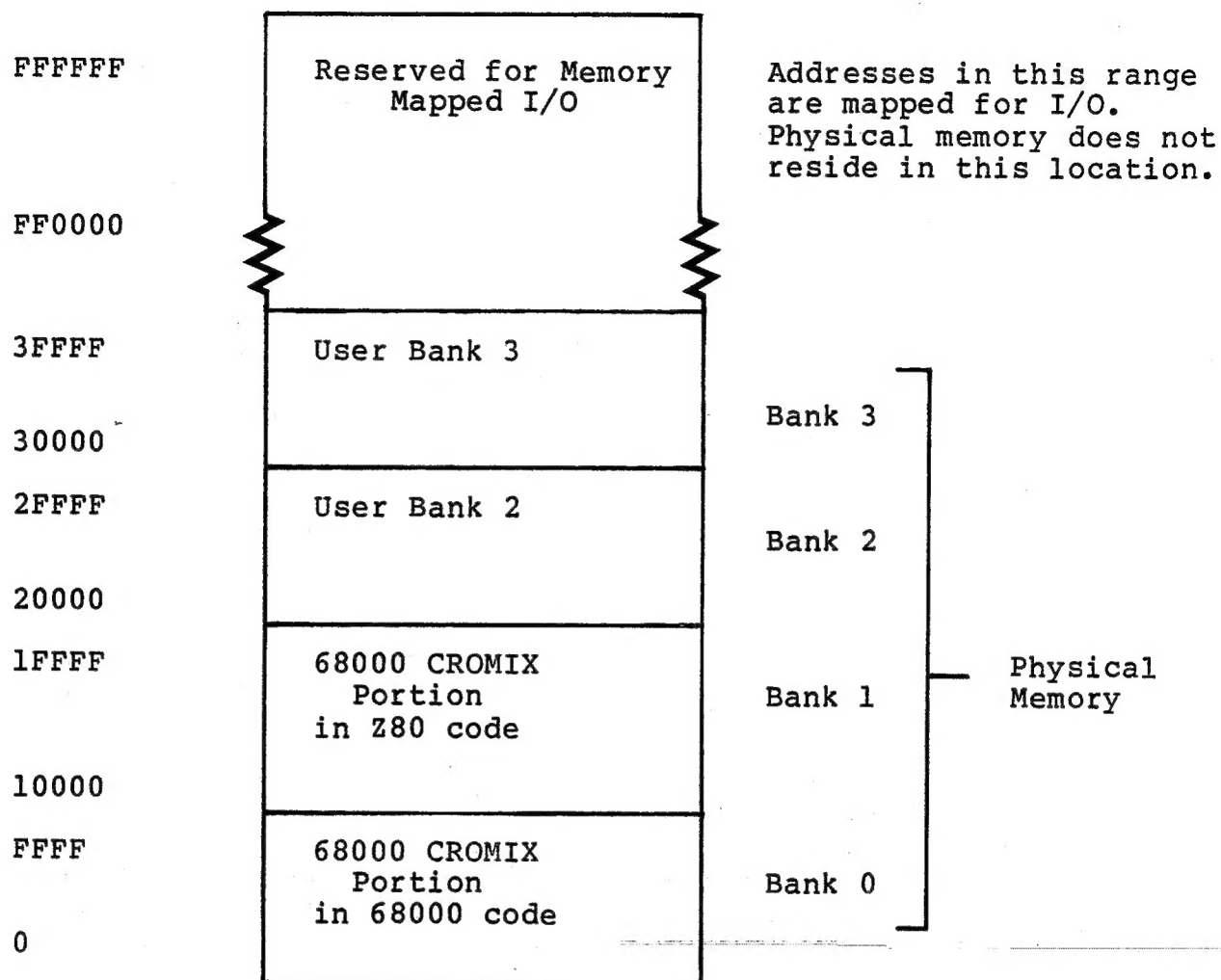
68000 Cromix memory is divided into two parts, **user memory** and **system memory**. The system memory is further divided into three parts: 64K bytes which contain the portion of Cromix software written in 68000 code, 64K bytes which contain the Z80 portion of the software, and the upper 64K bytes which provide memory mapped I/O for the 68000. User memory is divided into 64K byte sections one for each user.

Memory allocation for Z80 jobs is the same as with Z80 Cromix software. Jobs for the 68000, however, may be as large as available memory will allow. The number of available shell buffers, which depend on the options selected in generating the operating system, may be less than the number available with the Z80 Cromix software.

The 68000 generates 24 bit addresses and can directly address any portion of memory. The Z80 outputs 16 bit addresses, which comprise the low order 16 bits of 24 bit memory addresses. The upper eight bits of this address are contained in an address latch on the DPU board. The contents of this latch are set to the high order eight bits of the last op-code fetched by the 68000. This is the op-code that initiates Z80 processing.

The following is a graphic representation of 68000 Cromix memory usage:

Cromemco Software Update Service Note  
68000 Cromix version 20.05



## **68000 Cromix System Calls**

The same set of system calls available under the Z80 Cromix Operating System are available under the 68000 Cromix Operating System, with the exception of nine calls which remain to be implemented in the next version. System calls used by the 68000 are identical in function to corresponding Z80 system calls. The **Cromix Instruction Manual** describes the function of each call. Since these calls are being implemented on the 68000, however, their use of 68000 registers needs to be defined.

Upon encountering an error, the system call sets the carry flag and deposits an error code in register D0. The following table lists each 68000 system call, its call number, its calling parameters, and its return parameters. An asterisk following the name of a call indicates that the call has not yet been implemented.

Cromemco Software Update Service Note  
68000 Cromix version 20.05

System Call	System Number	Calling Parameters	Return Parameters
_alarm	43H	D3 = number of seconds	
_caccess	27H	D1 = channel D2 = access mask	
_cchstat *	23H	D1 = channel D2 = status type D3 = new value A1 -> buffer	
_chdup	0AH	D1 = existing channel	D2 = duplicate channel
_chkdev	07H	D2 = type of device D3 = major device # D4 = minor device #	
_clink	25H	D1 = channel A1 -> new pathname	
_close	0BH	D1 = channel	
_create	08H	D2 = access mode D3 = exclusive mode A0 -> pathname	D1 = channel
_cstat *	21H	D1 = channel D2 = desired info A1 -> buffer	D3 = Return value
_delete	06H	A0 -> pathname	
_error	1CH	D0 = error number D1 = channel Other registers as returned by the call that generated the error	
_exchg	0CH	D1 = channel D2 = channel	

Cromemco Software Update Service Note  
68000 Cromix version 20.05

System Call	System Number	Calling Parameters	Return Parameters
_exec *	4CH	A0 -> pathname A1 -> argument list	
_exit	46H	D3 = termination status	
_faccess	26H	D2 = access mask A0 -> pathname	
_fchstat *	22H	D2 = status type D3 = new value A0 -> pathname A1 -> buffer	
_fexec *	4BH	D1 = signal mask D2 = signal values A0 -> pathname A1 -> argument list	
_flink	24H	A0 -> old pathname A1 -> new pathname	
_fshell *	48H	D1 = signal mask D2 = signal values A0 -> pathname A1 -> argument list	
_fstat *	20H	D2 = desired info A0 -> pathname A1 -> buffer	D3 = return value
_getdate	30H		D0 = day of the week D1 = year D2 = month D3 = day of the month
_getdir	02H	A0 -> buffer	
_getgroup	36H	D2 = id type	D3 = group id
_getmode	12H	D1 = channel D2 = mode type	D3 = return value
_getpos	10H	D1 = channel	D3 = file position
_getprior	38H		D3 = priority number

Cromemco Software Update Service Note  
68000 Cromix version 20.05

System Call	System Number	Calling Parameters	Return Parameters
_getproc	3AH	D3 = process id	
_gettime	32H		D1 = hour D2 = minute D3 = second
_getuser	34H	D2 = id type	D3 = user id
_indirect	51H	D0 = call number Registers are used according to call number	Registers are used according to call number.
_kill	41H	D2 = signal type D3 = process id	
_lock	3EH	D2 = lock type D3 = lock length A0 -> lock sequence	
_makdev	00H	D2 = type of device D3 = major device # D4 = minor device # A0 -> pathname	
_mkdir	01H	A0 -> pathname	
_mount	04H	D1 = type of access A0 -> dummy pathname A1 -> device pathname	
_mult	MULU	D1, D0	
_open	09H	D2 = access mode D3 = exclusive mode A0 -> pathname	D1 = channel
_pause	44H		
_pipe	0EH		D1 = input channel D2 = output channel
_printf	1BH	D1 = channel A0 -> control string arguments on stack	

Cromemco Software Update Service Note  
68000 Cromix version 20.05

System Call	System Number	Calling Parameters	Return Parameters
_rdbyte	16H	D1 = channel	D0 = byte
_rdline	18H	D1 = channel D3 = maximum bytes A0 -> buffer	D3 = bytes read
_rdseq	14H	D1 = channel D3 = maximum bytes A0 -> buffer	D3 = bytes read
_setdate	31H	D1 = year D2 = month D3 = day of the month	
_setdir	03H	A0 -> pathname	
_setgroup	37H	D1 = type to change D2 = new id type D3 = new group id	
_setmode	13H	D1 = channel D2 = mode type D3 = new value D4 = mask	old value
_setpos	11H	D1 = channel D2 = mode D3 = file pointer	
_setprior	39H	D3 = priority number	
_settime	33H	D1 = hour D2 = minute D3 = second	
_setuser	35H	D1 = type to change D2 = new id type D3 = new user id	
_shell	*	49H	A1 -> argument list
_signal	*	40H	D2 = type of signal A0 -> execution addr.
_trunc	0DH	D1 = channel	

Cromemco Software Update Service Note  
68000 Cromix version 20.05

System Call	System Number	Calling Parameters	Return Parameters
_unlock	3FH	D2 = lock type D3 = lock length A0 -> lock sequence	
_unmount	05H	D2 = eject flag A0 -> device pathname	
_update	52H		
_version	55H		D3 = version number
_wait	45H	D2 = conditional flag D3 = process id	D1 = process term. D2 = system term. D3 = child pid
_wrbyte	17H	D0 = byte D1 = channel	
_wrline	19H	D1 = channel A0 -> buffer	D3 = bytes written
_wrseq	15H	D1 = channel D3 = byte count A0 -> buffer	D3 = bytes written

### TEMPORARY RESTRICTIONS ON THIS VERSION

The following capabilities are not supported by this version of the 68000 Cromix Operating System, but will be added in succeeding versions:

- Color Graphics**      The addressing of color graphics memory boards currently assumes horizontal memory organization and must be reconfigured.
- Default**              This Z80 Cromix command setting default parameters for automatic login is not implemented in this version.
- KSAM**                 The KSAM package has not yet been adapted to run under the 68000 Cromix Operating System.
- Shell Buffers**        Currently, there are a limited number of shell buffers available, the actual number depending on how the operating system is generated. Succeeding versions will have more shell buffers.
- System calls**        The following system calls are not implemented in this version:
- Exec
  - Fexec
  - Shell
  - Fshell
  - Cstat
  - Fstat
  - Cchstat
  - Fchstat
  - Signal

### VERSION NUMBER SUMMARY

Files in /	Version	
cromix.iop.sys	20.05	-new-
cromix.sys	20.05	-new-

Files in /bin	Version
access.bin	00.06
backup.bin	00.08
blink.bin	00.13
boot.bin	00.02
cdoscopy.bin	00.15

Cromemco Software Update Service Note  
68000 Cromix version 20.05

chowner.bin	00.06	
cmpasc.bin	00.05	
compare.bin	00.07	
copy.bin	00.10	
cptree.bin	00.07	
day.bin	01.02	
dcheck.bin	00.12	
deltree.bin	00.03	
dump.bin	00.10	
ecc.bin	00.02	-new-
echo.bin	00.05	
ed.bin	01.35	
find.bin	00.07	
free.bin	00.09	
group.bin	00.01	
h.bin	00.04	
help.bin	00.04	
icheck.bin	00.15	
idump.bin	00.06	
init.com	02.76	-new-
input.bin	01.00	
l.bin	00.11	
mail.bin	00.06	
makdev.bin	00.07	
makfs.bin	00.13	
maklink.bin	00.04	
match.bin	00.03	
mode.bin	01.12	
mount.bin	00.13	
move.bin	00.09	
msg.bin	00.08	
ncheck.bin	00.09	
passwd.bin	00.09	
patch.bin	00.03	
priv.bin	00.07	
restore.bin	00.05	
root.bin	00.02	
screen.bin	01.35	
sim.bin	00.34	
sort.bin	00.06	
spool.bin	00.12	
tee.bin	01.02	
testinp.bin	01.01	
time.bin	00.07	
unmount.bin	00.11	
usage.bin	00.06	
version.bin	00.09	
wboot.bin	00.09	
who.bin	00.06	

Cromemco Software Update Service Note  
68000 Cromix version 20.05

Files in /dev	Version
iop/cromix.iop	11.11
iop/ioprun.bin	03.00

Files in /etc	Version
fdboot	00.09
login.bin	00.02
sfdboot	00.09

Files in /gen	Version	
crogen68.bin	00.00	-new-
default.bin	00.01	